Evaluation of four search systems of Finnish Bible

Hurskainen, Arvi

SALAMA - Swahili Language Manager
2021-03-01

Hurskainen , A 2021 ' Evaluation of four search systems of Finnish Bible ' Technical Reports on Language Technology , no. 69 , SALAMA - Swahili Language Manager , Helsinki . < http://www.njas.helsinki.fi/salama/evaluation-of-four-search-systems-of-finnish-bible.pdf >

http://hdl.handle.net/10138/330099

Technical Reports on Language Technology
Report No 69, 2021
http://www.njas.helsinki.fi/salama

Evaluation of four search systems of Finnish Bible[1]

Arvi Hurskainen
Department of Languages
FIN-00014 University of Helsinki, Finland
arvi.hurskainen@helsinki.fi

Abstract

Information retrieval has been an important research area since the beginning of digital technology. The first search tools made use of direct mapping of the search key with the text. This method was ineffective, and more advanced solutions were needed. One solution was to use regular expressions in formulating the search key. However, words in languages, especially heavily inflecting languages, have such a vast number of forms, that the formulation of the search keys with regular expressions becomes cumbersome, and often impossible. There was a need for such an approach, where the target text is analysed and disambiguated, and the search is directed to this enriched text form. But even this is not enough. We must be able to form such disambiguated search keys, that we get only those words that we intend, and nothing else. Only when the target text is disambiguated, and also the search key is disambiguated, we can achieve precise search results.

Here I will compare four search systems of the Finnish Bible. Two of the systems use direct string search, one uses disambiguated text but non-disambiguated search keys, and one uses disambiguated text and disambiguated search keys.

Key Words: *information retrieval, language analysis, Bible search.*

1 Introduction

The search systems compared in this report are the search systems of Suomen Pipliaseura (RAAMATTU.FI), Raamattu.uskonkirjat.net (Uskonkirjat), Koivuniemen Raamattuhaku (Koivuniemi), and Salama. The first one has a search system, which makes use of analysed and disambiguated text. The two next ones use the direct string search, and the last one uses analysed and disambiguated text and disambiguated search keys.

First, I will describe each of these systems from the viewpoint of the user. I do not have access to the source code of the systems, except for Salama, for which I have written the code. Yet I hope that the essential features will be described adequately.

In the second phase I test the performance of each system with a set of search tasks. Finally, I make a statistical analysis of one of the search tasks. The results will be evaluated.

1.1 Raamattu.uskonkirjat.net (Uskonkirjat)

I cannot find a proper name for this search system. The web page[2] uses the net address as its name. The page has several Bible translations in various languages. It also has original biblical texts and other sources. Relevant to our task are the 1933/38 and 1992 Bible translations. The introductory text tells that the page is intended mainly for research purposes.

Immediately aside the search tab, there is the tab for search instructions. They contain advise on how to do such searchers as: simple string search, either/or search, partial word search, search of a phrase, and combined searches. The titles do not fully match with contents. Yet information is clear and useful.

An irritating feature is that the info page disappears as soon as the cursor is moved away from the info page and the mouse is clicked. It cannot be opened from the tab, because it is already open under the main page. When the info page is closed, it can be opened again.

The search keys have their own syntax.

The simple single-word search returns precisely similar whole words, not such words that have a partial match with the search key.

The either/or search is implemented simply by writing words after each other, with a space in between.

The NOT option is implemented by placing a minus sign '-' in front of the word. The MUST option is implemented by placing a plus sigh '+' in front of the word.

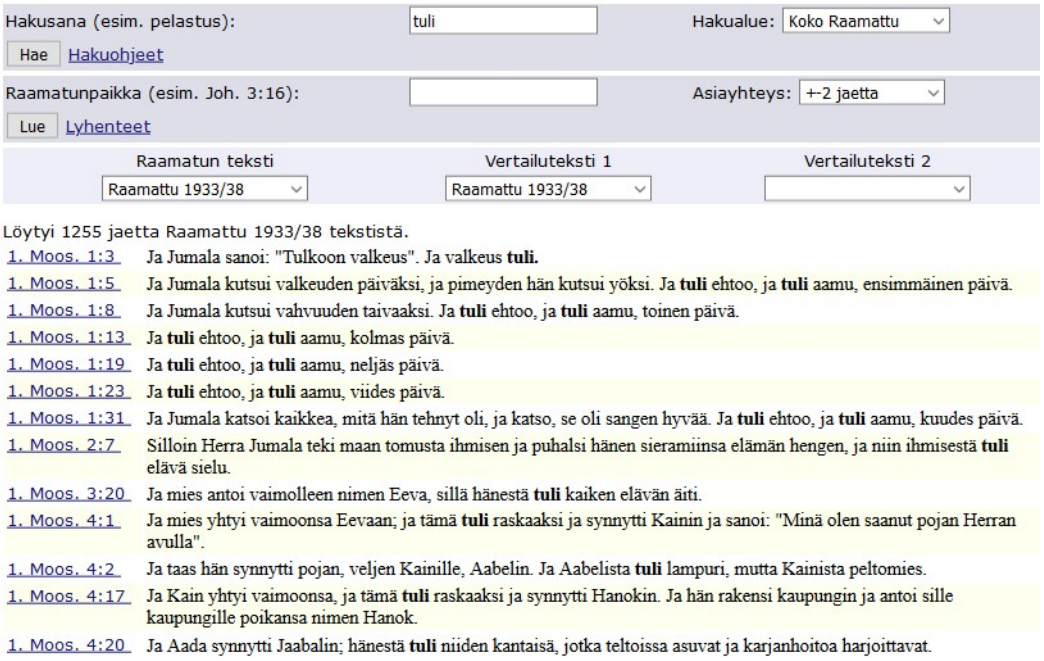
The partial key is implemented by placing the Kleene star after the first part of the word. Words starting with the search key will be found. There is no corresponding option for omitting the beginning part from the search key.

The phrase search option is implemented by placing double quotes around the phrase.

Various search styles can also be combined into the same search key.

The view of the Uskonkirjat web page is below.

[2] https://raamattu.uskonkirjat.net/servlet/biblesite.Bible



1.2 Koivuniemen Raamattuhaku (Koivuniemi)

This search system[3] contains two search environments, quick search, and extended search.

The quick search environment has only one search field, and it is intended for simple searches of individual words, phrases, or even whole sentences. It simply looks for direct matches in text and returns those lines (in this case Bible verses) that match with the search key. Also, regular expressions can be used in formulating more complex search keys. This is an advantage, although it is doubtful whether most users are willing to learn the syntax of regular expressions.

The extended search environment has three search fields for defining various criteria for search. Whereas in simple search, the string in the search window is interpreted as a single continuous sequence of characters and spaces, words in extended search are interpreted as separate units, and the relation of the words is defined. In one window, all words and phrases written in the window must be present in the verse, so that it will be returned. The second window is for writing either/or search conditions. The third window is for excluding words or phrases.

While the Boolean operators are often implemented by using operators such as AND, OR, and NOT, here they are implemented by using separate windows instead of operators. The solution looks logical, and from the user perspective it is clear. Especially for constructing complex search tasks the solution is excellent.

[3] https://www.koivuniemi.com/raamattu

The system has also a facility to define the target text. Even individual Bible chapters can be selected.

Also, the Bible translation can be selected. The latest official translation of 1992 is missing from the page. This is a pity, because the comparison of the systems can be made only by using the same translation. Three of the four systems have implementation of the 1992 translation. Because all four have implementations on the translation of 1938, I will use this translation on comparison.

The view of the Koivuniemi web page is below.

1.Mooseksen kirja:
1:3 Ja Jumala sanoi: "Tulkoon valkeus." Ja valkeus **tuli**.
1:5 Ja Jumala kutsui valkeuden päiväksi, ja pimeyden hän kutsui yöksi. Ja **tuli** ehtoo, ja **tuli** aamu, ensimmäinen päivä.
1:8 Ja Jumala kutsui vahvuuden taivaaksi. Ja **tuli** ehtoo, ja **tuli** aamu, toinen päivä.
1:13 Ja **tuli** ehtoo, ja **tuli** aamu, kolmas päivä.
1:19 Ja **tuli** ehtoo, ja **tuli** aamu, neljäs päivä.
1:23 Ja **tuli** ehtoo, ja **tuli** aamu, viides päivä.
1:31 Ja Jumala katsoi kaikkea, mitä hän tehnyt oli, ja katso, se oli sangen hyvää. Ja **tuli** ehtoo, ja **tuli** aamu, kuudes päivä. >> **Jakeen lisätiedot**
2:1 Niin **tuli**vat valmiiksi taivas ja maa kaikkine joukkoinensa.
2:7 Silloin Herra Jumala teki maan tomusta ihmisen ja puhalsi hänen sieraimiinsa elämän hengen, ja niin ihmisestä **tuli** elävä sielu.

1.3 Suomen Pipliaseura (RAAMATTU.FI)

RAAMATTU.FI[4] has search facilities to the 1938 and 1992 translations. The search environment is simple. It contains a search window for typing the search key, and a tab for executing the search. One cannot find any instructions on how to use the system, which is surprising. The info tab should be next to the search window. One can learn the system with try-and-error method.

A single word returns the verses, which match with the search key. The direct matches are also marked with dark colour. If the match is not direct but a different form of the same word, no marking appears on the hit.

The system also finds such verses, where the entered word appears in some other inflected form. This is something new, which is not present in the other two systems described above. The search key can be also in an inflected form, provided that this form appears in the Bible translation. If it does not appear in the target text, nothing will be returned.

The search system makes use of the analysed and disambiguated Bible text. In this process, each inflected form of a word can be associated to its base form, and the search is done on the basis of these mappings.

The solution is a big step forward in handling languages with rich inflection, such as Finnish. The solution itself does not solve all search problems. Its reliability depends very much on the accuracy of the disambiguation. A general analysis and disambiguation system as such does not suit to Bible texts, especially on the part or many proper names.

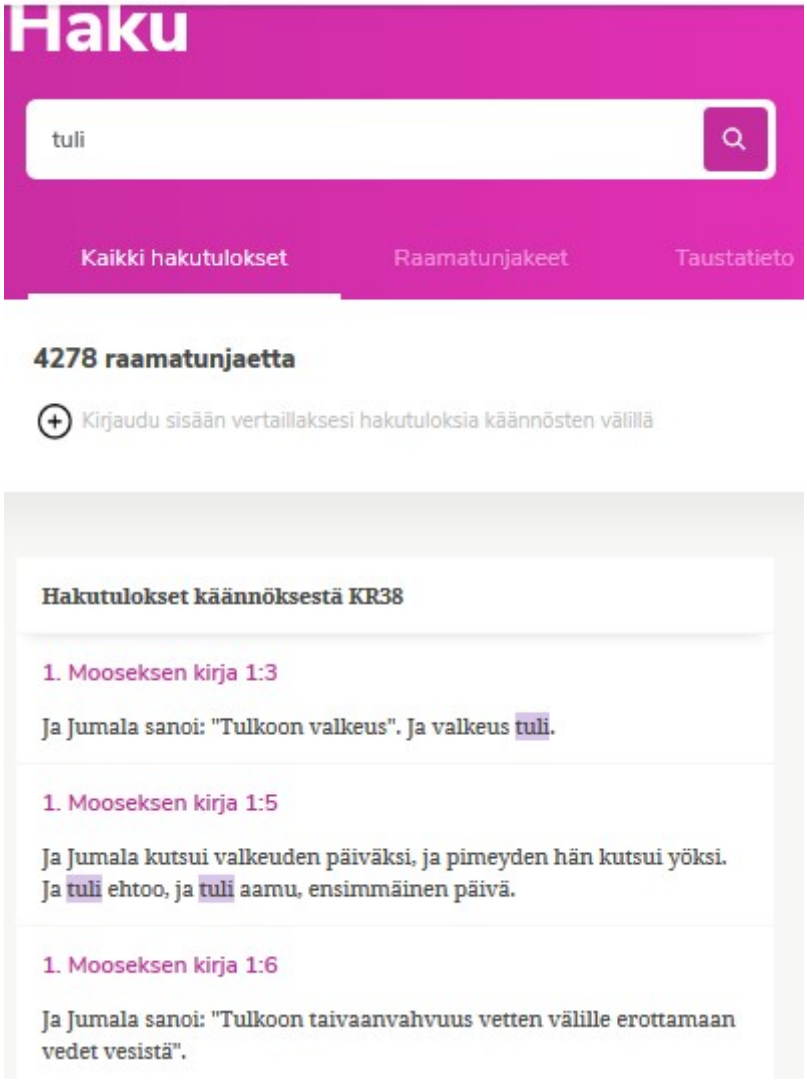
Although the search system of RAAMATTU.FI is poorly implemented, it has one facility that the other two systems are missing - the understanding that a language can also be represented in a more structured way, through analysis.

How reliable the system is will be discussed later in the test section.

In all, the search system of RAAMATTU.FI is a step forward in search accuracy. But it still has one serious weakness. In it, unique search keys cannot be formulated. Especially important would be the possibility to define the part-of-speech category of the search key. We come to this question blow.

The view of the RAAMATTU.FI web page is below:

[4] https://raamattu.fi/



1.4 Salama search system

While the three search systems described above are publicly available search environments, the Salama system[5] is so far on a private server, and access to it is limited. Yet its performance can be tested in a browser. There is no specific web page for Bible search. The search engines are part of many other applications, currently under the tab 'Tagger'. The info page gives detailed instructions on how to use each application, including the Bible search systems.

[5] 77.240.23.241/tagger

In Salama system, search is targeted to such a text format, where each surface word is followed by its bae form plus the part-of-speech code attached to it. In this paper I call it *rich text format*. An example of such format is in (1).

(1)

```
1Moos_1:1 Alussa {alku_N} loi {luoda_V} Jumala {Jumala_ERISN} taivaan
{taivas_N} ja {ja_CC} maan {maa_N}.
1Moos_1:2 Ja {ja_CC} maa {maa_N} oli {olla_V} autio {autio_A} ja {ja_CC}
tyhjä {tyhjä_A}, ja {ja_CC} pimeys {pimeys_N} oli {olla_V} syvyyden
{syvyys_N} päällä {päällä_POST}, ja {ja_CC} Jumalan {Jumala_ERISN} Henki
{Henki_ERISN} liikkui {liikkua_V} vetten {vesi_N} päällä {päällä_POST}.
```

In Salama, there are search applications for the 1938 and 1992 Bible, and for the Biblical apocrypha in Finnish.

There are two types of search engines for each of these three text collections.

In one system, the key word is first analysed, and the result is converted to the final search key. For example, if the word *kirjoissaan* is entered into the search box, it will be converted to the form `{kirja_N}`, and the actual search will be done using this form as a key.

In another system, there is no prior analysis. The search key is formulated according to need. This system allows many kinds of search possibilities.

The search can be targeted to surface text, whereby single words and phrases can be searched. Even whole verses can be used as search key. The search key must fully match with the text.

The system does not automatically convert capital letters to lower case letters. Using this solution, we can add precision to the search. If capital-initial words and lower-case words are searched, regular expressions can be used, such as `[Oo]tsasi`.

The real power of this system can be seen when we direct the search to the analysed form of the text. For example, the search key `{tuli_N}` finds all occurrences of the noun *tuli*. The search key `{tulla_V}` returns all the occurrences of the verb *tulla*. All the inflected forms will be found, because the base form plus its part-of-speech tag are displayed after the surface form of each word.

The full match of the search key with the form in text includes curly brackets around the string. They are the boundary marks of the word. If we omit the left bracket, also words with longer stem but with the same latter part will be found. The key can often be shortened. For example, the key `tuli_` finds all forms of the noun *tuli* in text, and no forms of the verb *tulla*.

If it is required that two or more words appear in the verse, this can be implemented using one of the reserved words, AND, JA, or NA in between the words. The words can be surface forms or analysed forms. The requirement is that the words in the verse must be in the same order as in the search key. This solution was made for increasing the accuracy of search.

The search key can be cut in the beginning or end of the word by placing the asterisk '*' for marking the cut point.

The view of the Salama web page is below.

2 Search methods

In this section I compare the search systems in relation to their ability to perform different kinds of search tasks.

2.1 Direct string search

Here I compare the search of individual words, word phrases, and whole sentences.

Uskonkirjat: In this application, single words can be searched by typing the word into the search box. For searching phrases, double quotes should be placed around the phrase. Upper-case letters and lower-case letters have the same value. No distinction is made between them.

Koivuniemi: Single words can be searched. Whole words can be written into the search box. Also partial words as key work, as far as the key matches part of the word. The key may match also inside the word, so that part of the word is outside the match on both sides. Phrases can be searched by simply typing the phrase into the search box. Capital letters and lower-case letters have the same value.

RAAMATTU.FI: Single words can be searched by typing any form of the word. All forms of the word will be retrieved. Direct matches appear in dark colour. Indirect matches are retrieved without marking the matched words. Phrases can be retrieved, and phrases with literal match are marked with dark colour. In addition to the phrases, also such verses will be found, which have all the words of the phrase in some form and in

any order. This makes the finding of phrases problematic. Partial words can be found by using an asterisk '*' at the end of the string. The beginning part of the word cannot be cut.

Salama: When a word in text is used as search key, the search is automatically done from surface text. When a single word is entered, all direct matches will be retrieved. Also such words will be retrieved, which have a direct match and something else in the word. The extra material can be in front or after the matched part, or in both. Whatever the type of match is, the whole word - not only the match - will be surrounded with some type of braces, depending on the type of match. If the search key fully matches with the found word, it will be surrounded with angle brackets '<' and '>'. If the key matches only to part of the word, the whole word will be surrounded with square brackets '[' and ']'. Also phrases of up to four words long can be retrieved. The found phrase will be surrounded with angle brackets. Currently the search cannot be longer, because the target text is enriched with base form after each surface word, and it is not an easy thing to match the surface string over the base forms. One solution for facilitating longer search strings would be to return the text into normal text in flight and then search from this text. But it is questionable whether there is real need for such search.

Salama is different from all other search systems in that the search can be directed to the base form. The base form has the format `{word_POS}`, where word is a base form of any word, and POS is a tag of some part-of-speech category. The combination of these two elements in search key makes the search accurate, provided that the target text is correctly disambiguated. If the search key includes an underscore '_', the search is automatically targeted to base forms. Also, partial keys can be used. For example, the key `_A}` will retrieve all adjectives.

To retrieve words that start with a certain string, an asterisk '*' at the end of the string can be used. In Salama, one can also cut the beginning part of the word by placing the asterisk '*' in front of the search key. All words that have the same end part as the key will be retrieved.

2.2 Search using Boolean operators

Boolean operators AND, OR, and NOT are implemented using various techniques.

Uskonkirjat: This application has an easy-to-use implementation of the Boolean operators. If a plus sign '+' is placed in front of the word, this word must be in the target verse. The minus sign '-' in front of the word means that the word is not allowed to be in the target verse. The OR operator is not implemented in the application, although the sub-title on the info page indicates it. In fact, the sub-title is misleading.

Koivuniemi: This application has a separate window for each of the Boolean operators. The AND operation is in one window. All words or search conditions written into this box must be satisfied in the target verse. The OR operation is in the second window. Alternative words or search conditions can be typed here. The NOT operation is in the third window. No word or search condition typed here may occur in the target verse. The application is clear and easy to use. There is also a short info above each search box.

RAAMATTU.FI: It is hard to figure out how this application has implemented Boolean operators, because no information can be found on the page of the application. Therefore, no detailed description of it can be given. By testing I have found that when typing more than one word into the search box, all such verses will be retrieved, which have all these words, in any order. Also inflected forms are counted. It is not possible to restrict the search to literal matches.

Salama: This application has implemented the Boolean operators AND and OR, but not the NOT operator. Because the search is targeted to the rich text format instead of normal text, the number of words in the search key is restricted to four. Salama has the special feature that surface forms and base forms can be used in the search key. Because by default the string matches to all such words in target text, which have the same string as the search key, for getting precise results, it is better to use search based on base forms. For example, the key `laki_ TAI evankeliumi_` returns all such verses, where one of these words occurs, including inflected forms. The left context in the search key is left open, and also such forms as `päälaki_N` and `suulaki_N` will be retrieved. We can mark the left boundary using the left curly brace '{' or dot '.' in front of the search word. Note that this only works when a base form is used as a key. The key `{laki_` or `.laki_` retrieves only the verses with the noun *laki*. The AND operator allows up to four words to be in the search key. In this application, the words must occur in the target verse in the same order as in the search key. The solution was made for increasing the accuracy of search. Also here, surface forms and base forms can be used in search key. Surface forms and base forms can also be mixed in the search key.

2.3 Search of partial words

Applications differ in how partial words can be searched. Below is a short description.

Uskonkirjat: This application has only one method for searching partial words. The Kleene star or asterisk at the end of the search key means that the target word may be the same as the search key, or it can be longer. The search key cannot be cut in the beginning of the word.

Koivuniemi: In simple search, all such words will be found, where the search key matches the whole word or part of it. The match may be also inside the word. When an asterisk is put after the search key, it has no effect. It means that the beginning part of the search word cannot be formulated. The match is not perfect. Also such words will be found, which have only partial mapping. This is strange.

RAAMATTU.FI: This application apparently does not have a possibility of cutting the search word. At least asterisk at the end of the search string does not work.

Salama: This application allows the search key to be cut at the end and in the beginning. An asterisk is used for this purpose. Note that if no asterisk is used, the search key matches any string in target text, even inside the word. But if you use an asterisk in the end or at the beginning, this forces the string to match precisely with the beginning part

or end part of the word. This solution makes the precise search of also the surface strings possible. The matched words (whole words) are surrounded with square brackets.

2.4 Search of phrases

Although the search of phrases was already mentioned above, here I compare them in a systematic way.

Uskonkirjat: This application has implemented the search of phrases in a simple way. When the set of search words is surrounded with double quotes, the string of words is interpreted as a continuous string of words, with no other words in between. This is a handy way for searching phrases.

Koivuniemi: The phrase is simply typed into the search box. The search key need not consist of whole words. The beginning and end parts of the phrase can be omitted.

RAAMATTU.FI: It is not known how phrases can be searched in this application.

Salama: Phrases of up to four words can be searched. The phrase should be typed as it is in text. Capital letters and lower-case letters are different. Regular expressions can be used for finding also sentence-initial phrases. Only whole words can be used.

3 Comparing the search system using search tasks

Below I will make a set of tests using the same search task for each system.

Task 1: Find all occurrences of the noun *tuli*.
Find all occurrences of the verb *tulla*.

Uskonkirjat

`tuli` 1255 occurrences, contains occurrences of the noun *tuli* and verb *tulla*
`tul*` 4874 occurrences, contains all words beginning with *tul*.

Koivuniemi

`tuli` 1944 occurrences, includes all words containing the string *tuli*.
`tul*` 14532 occurrences, includes all words containing the string *tul*. Also many non-searched hits are found.

RAAMATTU.FI

`tuli` 4278 occurrences, includes all words containing the verb *tulla* and some of noun *tuli*, and some other words. It is not possible to make a difference between nouns and verbs in this application.
`tul*` 5441 occurrences, includes all words that begin with *tul*.

Salama: `tuli_N` 355 occurrences
`tulla_V` 4406 occurrences

Salama is the gold standard, and we compare the performance of other systems with it. We see that the results with the same search key are very different. The search key *tuli* does not differentiate nouns and verbs, except for in Salama, which uses disambiguated search keys. Therefore, the search key finds also many non-nouns in all three systems. The system of RAAMATTU.FI finds also such forms, which are inflected forms of the word *tuli*, including nouns and verbs.

The attempt to find the verb *tulla* using the key `tul*` produces also many unwanted hits, but the accuracy is better than in searching for the noun *tuli*.
In all, only Salama performs the given task.

Task 2: Find the verb *vanhurskauttaa*.
Find the noun *vanhuskaus*.
Find the noun *vanhurskas*.
Find the adjective *vanhurskas*.

Uskonkirjat
`vanhurskas*` 118 occurrences, mostly nouns
`vanhurskaut*` 138 occurrences, nouns and verbs
`vanhurskau*` 321 occurrences, nouns and verbs
It is not possible to make difference between part-of-speech categories.

Koivuniemi
`vanhurskas*` 575 occurrences, also partial mappings occur
`vanhurskaut*` 322 occurrences, also partial mappings occur
`vanhurskau*` 575 occurrences, also partial mappings occur
It seems that when the asterisk is put at the end of the string, the last character in the string is cut off from the final search string. This is obviously a bug in the system. This increases the number of hits but at the same time causes overproduction.

RAAMATTU.FI
`vanhurskas*` 312 occurrences, includes also inflected forms
`vanhurskaut*` 29 occurrences, includes nouns and verbs
`vanhurskau*` 353 occurrences, includes nouns and verbs
Salama
`vanhurskas_N` 130 occurrences (nouns)
`vanhurskas_A` 150 occurrences (adjectives)
`vanhurskauttaa_V` 16 occurrences (verbs)
`vanhurskautus_N` 1 occurrence (noun)
`vanhurskaus_N` 319 occurrences (nouns)
Salama makes a difference between part-of-speech categories. Therefore, the result is accurate. Koivuniemi seems to produce abundantly hits, which makes accurate search difficult. In all, search systems have huge differences in search accuracy.

Task 3: Find the verses with the nouns *armo* and *rauha*.

Uskonkirjat
`+armo +rauha` 18 occurrences

Koivuniemi
`armo rauha` 22 occurrences

RAAMATTU.FI:
`armo rauha` 21 occurrences

Salama
`armo_ AND rauha_` 19 occurrences

The search task was very simple, because the words are in base form in Bible and they have no ambiguity. Therefore, the result was almost the same in all systems.

Task 4: Find the pronoun *se* and the conjunction *sillä*. The task is not easy, because many occurrences are ambiguous, and it is difficult to formulate the search key.

Uskonkirjat:
`se` 2684 occurrences
`sillä` 3804 occurrences

Koivuniemi:
`se` 18808 occurrences
`sillä` 3897 occurrences

RAAMATTU.FI:
`se` 13379 occurrences
`sillä` 13379 occurrences
It seems that this application interprets the conjunction *sillä* as an inflected form of the pronoun *se*, because both searches give the same result.

Salama:
`{se_PRON` 10270 occurrences
`{sillä_CONJ` 4010 occurrences

The pronoun *se* has a bigger variation than the conjunction *sillä*, because it has many inflected forms and systems have differences in how to handle inflected forms. Yet all but Salama mix the pronouns and conjunctions.

Task 5: Find verses that have the verb *kuulla* and *nähdä*.

Uskonkirjat:
`kuulla AND nähdä` 12 occurrences

Koivuniemi:
`kuulla nähdä` 13 occurrences

RAAMATTU.FI:
`kuulla nähdä` 33 occurrences

Salama:
`kuulla_V AND nähdä_V` 52 occurrences
`nähdä_V AND kuulla_V` 57 occurrences

First two applications have almost the same result, because they search on the basis of the surface form. The third one is supposed to find also inflected forms, but it finds only 33 of the total 109. Salama gives separate results for both orders of the search words.

Task 6: Find verses that have the verb *syödä* and *juoda*.

Uskonkirjat:
`syödä AND juoda` 13 occurrences

Koivuniemi:
`syödä juoda` 20 occurrences

RAAMATTU.FI:
`syödä juoda` 134 occurrences

Salama:
`syödä_V AND juoda_V` 128 occurrences
`juoda_V AND syödä_V` 13 occurrences

The result is interesting. It shows that RAAMATTU.FI and Salama have almost identical results, because both find also inflected forms. The first two find only a small part of verses. In this test, RAAMATTU.FI performs well, because the search keys are not ambiguous.

Task 7: Find verses with the surface form *pojalla* and the verb *olla*.

Uskonkirjat:
`pojalla AND on` 6 occurrences

Koivuniemi:

`pojalla on` 10 occurrences

RAAMATTU.FI:
`pojalla on` 1592 occurrences

Salama:
`pojalla AND .olla_V` 9 occurrences

All applications, except for RAAMATTU.FI, have fairly similar results. The exceptional result of RAAMATTU.FI is due to the mode of operation. It counts all the forms of the noun *poika* and of the verb *olla*, and all such verses are retrieved, where a form of these two words appear, in any order. In Salama, it is possible to search for the surface form *pojalla* and any inflected form of the verb *olla*.

Task 8: Find verses with any form of the pronoun *hän* and the verb *olla*.

Uskonkirjat:
`hän AND on` 1975 occurrences

Koivuniemi:
`hän on` 6271 occurrences

RAAMATTU.FI:
`hän on` 7542 occurrences
Salama:
`.hän_PRON AND .olla_V` 4880 occurrences
`.olla_V AND .hän_PRON` 3789 occurrences

In this search task, RAAMATTU.FI and Salama have similar results. This is largely due to the fact that both find inflected forms of words, and the search keys are not ambiguous. The first two applications find only whole surface words and therefore have low scores. The second system finds also matches inside words, and therefore the scores are higher.

4 Statistical evaluation of the search systems

On the basis of the tests above, we have got a general view of the performance of the systems, but we do not know where exactly they succeeded and where they failed. The numbers themselves do not reveal the number of true and wrong hits. Below I will compare the findings of each system verse by verse, keeping the results of Salama as gold standard.

The test required preparatory work, because each search system behaves differently. Koivuniemi, for example, wraps the results into sections, and each section must be opened separately for finding all hits. Also the encoding of verses is different in each application. All these versions were converted into Salama-like encoding, so that comparison became possible.

The detailed test was made of the Task 1 above, that is, the search of the noun *tuli* and the verb *tulla*. The results are in Table 1.

Table 1. Performance of the search systems.

	tuli			**tulla**		
System	*Correct*	*Wrong*	*Missing*	*Correct*	*Wrong*	*Missing*
RAAMATTU.FI	91	3746	273	3466	383	941
Uskonkirjat	66	1190	289	1197	69	3209
Koivuniemi	79	1849	275	1825	113	2580
Salama	**355**	0	0	**4406**	0	0

Key:
Correct = the hit appears also in Salama results
Wrong = the hit does not appear in Salama results
Missing = the hit appears in Salama results but not here

We see that when we search the noun *tuli*, the success rate with all three systems is very low. RAAMATTU.FI is slightly better than the others, although it should find also the inflected forms. The low rate with this search task is mainly due to the fact that when the search key *tuli* was used, it was able to find only a small part of the noun *tuli*.

When we compare the results of the three search systems with the search results of the verb *tulla*, the result is much better. RAAMATTU.FI finds correctly most of these verbs, although it does not specify that they are verbs. The other two have lower scores, and the number of correct hits is lower than the number of correct missing hits.

5 Discussion and conclusion

Although the aim of this evaluation of the search system was to compare their performance, it finally turned out that Salama was in the role of gold standard, because its performance was nearly 100 percent in terms of recall (all intended hits were found) and precision (only intended hits were in the result). All other systems were very far from perfect.

The solution made in Salama, that the text is converted to such format that each word in text has also its base form plus part-of-speech code displayed, is just one example of how analysed text can be made use of for increasing accuracy of search. The result of the analysis process also includes many other kinds of tags, such as syntactic tags, semantic tags, and tags encoding inflection. Any of these tags can be imported into the rich text, so that they can be made use of in search. Professional corpus retrieving systems, such as Korp[6] in the META-SHARE of EU, and Sketch Engine[7], originally developed by Adam Kilgarriff but later integrated into the EU repository of search engines, are perhaps too detailed for Bible search, especially for ordinary users. Yet they are illuminating

[6] https://korp.csc.fi/korp/
[7] https://www.sketchengine.eu/corpora-and-languages/

examples of how structured information in language can be made use of for increasing accuracy of search.

All four Bible search systems have the traditional string search mechanisms. RAAMATTU.FI finds also inflected forms, but it does not find all of them. If it would do it, it should not have any missing hits in the test. Also, non-ambiguous search keys cannot be formulated in it.

Salama has the possibility to search also on the basis of the base form, which makes the search fully accurate. It finds all what is wanted and nothing else. This is true provided that the target text is disambiguated correctly. In this test, the target text was manually checked in relation to test words and disambiguation errors were corrected. Therefore, Salama results can be treated as gold standard.

Salama has also an application, where the search word is first analysed and converted into the combination of the base form and its part-of-speech code. The search is then done on the basis of this form. In other words, it always searches on the basis of the base form. This application is user-friendly, but its search possibilities are limited.

The systems differ in how the long results are printed. Salama and Koivuniemi print the whole result as one piece. RAAMATTU.FI prints gradually, and it is difficult to copy the result, if it is long. Uskonkirjat wraps the result into separate sections, and it is very difficult to copy the whole result.